

실습) 2개의 Time 인스턴스를 생성하여 인스턴스간의 산술연산(+, -) 결과와 비교연산 결과를 출력하는 프로그램을 짜시오

시간을 나타내는 Time 클래스를 정의한다.

- 시는 0~23, 분은 0~59, 초는 0~59 입력

1) 초기화(생성자) 특수 메소드 정의 하시오

- init 특수 메소드 정의

2) +, - 연산자를 중복 정의 하시오

- add, sub 특수 메소드 정의

- 초로 변환해서 연산

- /, % 사용해서 시/분/초로 변환해서 저장

- + 연산시 24시가 넘을 경우, - 연산시 0시 이전이 될 경우

3) <, >, == 연산자를 중복 정의 하시오

- lt, gt, eq 특수 메소드 정의

- 초로 변환해서 연산

4) print 함수를 중복 정의 하시오

- repr 특수 메소드 정의

class Time:

초기화(생성자) 특수 메소드

def __init__(self, hour, min, sec, day=0):

 self.day = day

 self.hour = hour

 self.minute = min

 self.second = sec

add 특수 메소드

def __add__(self, other):

 total = (self.hour*60*60 + self.minute*60 + self.second) +

 (other.hour*60*60 + other.minute*60 + other.second)

 max = 24*60*60

 if (total >= max):

 _____ # 두 문장

 _____ # hour 값 저장

 _____ # min 값 저장

 _____ # sec 값 저장

 return _____ # 결과 객체 생성후 리턴

sub 특수 메소드

def __sub__(self, other):

 # 코딩 - total 0이 음수일 경우 고려

lt 특수 메소드

def __lt__(self, other):

 # 코딩

```
# eq 특수 메소드
def __eq__(self, other):
    # 코딩

# gt 특수 메소드
def __gt__(self, other):
    # 코딩

# repr 특수 메소드
def __repr__(self):
    # 코딩 - O 일 OO 시 OO 분 OO 초 입니다 형태로 출력

hour = int(input("시 입력"))
min = int(input("분 입력"))
sec = int(input("초 입력"))
first_time = Time(10, 10, 10)
second_time = Time(hour, min, sec)
add_time = first_time + second_time
sub_time = first_time - second_time
print("두 시간의 합은")
print(add_time)
print("두 시간의 차는")
print(sub_time)
if (first_time > second_time):
    print("첫번째 시간이 늦은 시간입니다")
if (first_time < second_time):
    print("첫번째 시간이 이른 시간입니다")
if (first_time == second_time):
    print("두 시간이 같은 시간입니다")
```